Pertemuan 2

Editor vi

Objektif:

1. Praktikan menguasai apa yang dimaksud dengan editor vi.
2. Praktikan mengetahui apa saja yang termasuk dalam utilitas aritmatika.
3. Praktikan mengetahui perintah-perintah yang terdapat utilitas teks.

P2.1. Teori

Editor vi

Posisi vi dibedakan menjadi dua, yaitu:

1. Mode perintah.

Setiap pemakai masuk ke dalam editor vi, langsung pada mode perintah. Pada kondisi ini, pemakai dapat melakukan:

a) menggerakkan kursor.
b) menghapus teks.
c) mengubah teks.
d) mencari kata.
e) keluar dari vi ke shell.

2. Mode Penyisipan.

a) untuk berpindah ke mode penyisipan dari mode perintah, dilakukan dengan mengetikkan huruf i.
b) untuk kembali ke mode perintah, dilakukan dengan menekan tombol <ESC>.

Ada beberapa cara untuk keluar dari editor vi (harus pada mode perintah), yaitu:

1) **:wq <enter>**, digunakan untuk menyimpan teks.
2) **:x <enter>**, digunakan untuk menyimpan teks.
3) **:zz** , digunakan untuk menyimpan teks.
4) **:q! <enter>**, keluar dari vi tanpa menyimpan.

Mode-mode Perintah pada Editor vi

Beberapa perintah yang digunakan adalah :

1. Menggerakkan Kursor

Perintah yang digunakan adalah sebagai berikut:

Tabel 2.1 Perintah menggerakkan kursor

Perintah	Fungsi
h	satu karakter ke kiri
l	satu karakter ke kanan
j	satu karakter ke atas
k	satu karakter ke bawah

w	ke awal kata berikutnya
e	ke akhir kata
b	ke awal kata sebelumnya
^	ke karakter bukan spasi atau tab yang pertama
0	ke awal baris
$	ke akhir baris
|	ke awal baris
n	ke kolom n
-	ke karakter non-spasi pertama pada baris sebelum posisi kursor
<enter>	ke karakter non-spasi pertama pada baris berikutnya
(	ke awal kalimat
)	ke akhir kalimat
{	ke awal paragraph
}	ke akhir paragraph
%	ke pasangan dari ({[atau]}) yang terletak pada posisi kursor

2. Menggerakkan Layar

   Perintah-perintah yang digunakan sebagai berikut:

   Tabel 2.2 Perintah menggerakkan layar

Perintah	**Fungsi**
z <enter>	menempatkan baris pada posisi kursor menjadi baris awal pada layar
z	menempatkan baris pada posisi kursor menjadi di tengah layar
z-	menempatkan baris pada posisi kursor menjadi baris terakhir pada layar
<ctrl-e>	layar bergerak naik satu baris
<ctrl-y>	layar turun satu baris
<ctrl-u>	tampilan bergerak turun setengah layar
<ctrl-d>	tampilan bergerak naik setengah layar
<ctrl-f>	tampilan bergerak naik satu layar penuh
<ctrl-b>	tampilan bergerak mundur satu layar penuh
nG	menuju baris ke-n
$G	menuju baris terakhir

3. Pembatalan atau Pengulangan Perintah

Tabel 2.3 Pembatalah atau pengulangan perintah

Perintah	Fungsi
u	membatalkan perintah terakhir yang dilakukan
U	membatalkan beberapa perintah yang telah dibuat pada satu baris (mengembalikan baris ke keadaan semula)
dot (.)	melakukan pengulangan perintah terakhir

4. Menyisipkan Teks

Perintah-perintah yang digunakan sebagai berikut :

Tabel 2.4 Perintah menyisipkan teks

Perintah	Fungsi
a	menambahkan teks mulai posisi sesudah kursor
i	menambahkan teks mulai posisi sebelum kursor
A	menambahkan teks setelah akhir baris
I	menambahkan teks dimulai pada posisi karakter bukan spasi pertama
o	menyisipkan baris kosong sesudah baris pada posisi kursor
O	menyisipkan baris kosong sebelum baris pada posisi kursor

5. Menghapus Teks

Perintah-perintah yang digunakan sebagai berikut :

Tabel 2.5 Perintah menghapus teks

Perintah	Fungsi
x	menghapus karakter pada posisi kursor
nx	menghapus n karakter dari posisi kursor
X	menghapus karakter di sebelah kiri kursor
nX	menghapus n karakter di sebelah kiri kursor
dd<enter>	menghapus baris yang ada pada posisi kursor
d <enter>	menghapus dua baris dimulai dari posisi kursor (sama dengan 2dd)
ndd	menghapus 13 baris dimulai pada posisi kursor (sama dengan

<enter>	dnd), jika jumlah baris sesudah posisi kursor kurang dari 13, maka penghapusan dibatalkan
d	menghapus teks dari awal baris hingga sebelum posisi kursor
dw	menghapus dari posisi kursor sampai akhir kata
dW	menghapus dari posisi kursor sampai akhir kata yang dipisah oleh spasi
db	menghapus dari awal kata sampai sebelum posisi kursor semula
dB	identik dengan db, kecuali pengertian kata di sini adalah yang dipisah oleh spasi
d$	menghapus dari posisi kursor hingga akhir baris
D	sama dengan d$
d(	menghapus dari awal kalimat hingga sebelum posisi kursor
d)	menghapus dari posisi kursor hingga akhir kalimat
d{	menghapus dari posisi kursor hingga akhir paragraf
d}	menghapus dari awal kalimat hingga sebelum posisi kusor

6. Mengganti Teks

Perintah-perintah yang digunakan sebagai berikut :

Tabel 2.6 Perintah mengganti teks

Perintah	**Fungsi**
r	mengganti tepat satu karakter pada posisi kursor adalah r (tanpa berada pada mode penyisipan)
nrq	mengganti n karakter dari posisi kursor dengan karakter q
R	mengganti karakter dimulai dari posisi kursor dan berhenti apabila pemakai menekan tombol <esc>
s	mengganti satu karakter dengan beberapa karakter lainnya sampai tombol <esc> ditekan
S	mengganti teks pada baris tempat kursor sedang berada
cw	penggantian sampai akhir kata
cW	penggantian sampai akhir kata yang dipisah oleh spasi
cb	penggantian dari awal kata sampai sebelum posisi kursor
cB	penggantian dari awal kata yang dipisah oleh posisi kursor
cc	penggantian pada baris tempat kursor berada
c	penggantian dari awal sampai sebelum posisi kursor
c$	penggantian dari posisi kursor hingga akhir baris

C	sama dengan c$
c(	penggantian dari awal kalimat sampai sebelum posisi kursor
c)	penggantian dari posisi kursor sampai akhir kalimat
c{	penggantian dari posisi kursor sampai akhir paragraf
c}	penggantian dari awal paragraf sampai sebelum posisi kursor
~	mengganti huruf kapital menjadi huruf kecil atau sebaliknya, pada posisi kursor
n~	mengkonversi n karakter mulai dari posisi kursor

Utilitas Aritmatika

1) expr

Merupakan utilitas yang biasa dipakai untuk melakukan perhitungan bilangan bulat dan operasi perbandingan . Operator yang digunakan:

Tabel 2.7 Operator expr

Operator	Keterangan
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
%	Sisa Pembagian
=	Sama dengan
\<	Kurang dari
\<=	Kurang dari atau sama dengan
\>	Lebih dari
\>=	Lebih dari atau sama dengan
!=	Tidak sama dengan
\\|	Operator or
\&	Operator and
:	Operator *match* (digunakan untuk mencocokan string operand kedua terhadap string operand pertama).

2) bc

Merupakan utilitas yang sangat bermanfaat untuk membantu melakukan perhitungan aritmatika, baik bilangan bulat maupun pecahan. Operator yang ada di antaranya:

Tabel 2.8 Operator bc

Operator	Keterangan
+	Penjumahan
-	Pengurangan
*	Perkalian
/	Pembagian
%	Sisa pembagian
++	menikan variable sebesar satu
--	menurunkan nilai

3) dc

Merupakan utilitas yang sama dengan bc, perbedaannya hanya pada bentuk notasi ekspresi aritmatikanya. Jika pada bc ekspresi aritmatika dinyatakan dengan operasi infix (operator berada di antara operand) maka pada dc dinyatakan dengan operasi postfix (operator berada di akhir dari dua operand).

Contoh:

Tabel 2.9 Perbedaan bc dan dc

bc (INFIX)	dc (POSTFIX)
2 + 3	2 3 +
a + b ^ c	a b c ^ +

Utilitas Teks

1) pg

Bentuk umum:

```
pg [ file…]
```

Merupakan perintah untuk menampilkan isi satu file atau lebih per layar.

Perhatikan aturan perintah sebagai berikut :

- Setiap data satu layar ditandai dengan simbol **:** pada bagian bawah layar.
- Tombol **<enter>** digunakan untuk berpindah layar.
- Untuk keluar dari pg, digunakan tombol **q <enter>**.
- Akhir file ditandai dengan **(EOF):** singkatan dari *end of file*.

2) more

Bentuk umum:

more [file…]

Digunakan untuk menampilkan isi satu file atau lebih per layar.

Perhatikan aturan pada perintah ini :

- Setiap data satu layar ditandai dengan simbol **-more** pada bagian bawah layar dan diikuti dengan prosentase data yang telah ditampilkan ke layar.
- Tombol **<enter>** digunakan untuk menggeser satu baris ke atas.
- Untuk berpindah layar digunakan tombol **<spasi>**.
- Untuk keluar dari more, digunakan tombol **q**.

3) head

Bentuk umum:

head [-n] [file **]**

Digunakan untuk mendapatkan baris yang merupakan bagian awal dari suatu file.

Option **-n** digunakan jika hanya n baris pertama dari file yang akan ditampilkan ke standar output, jika tidak diberikan maka hanya 10 baris pertama dari file yang akan ditampilkan.

4) tail

Bentuk umum:

tail [-n] [file **]**

Digunakan untuk mendapatkan bagian akhir dari suatu file, kebalikan dari head.

Option **-n** digunakan jika hanya n baris terakhir dari file yang akan ditampilkan ke standar output, jika tidak diberikan maka hanya 10 baris terakhir dari file yang akan ditampilkan.

Simbol **+** dapat digunakan untuk menggantikan simbol **-**, jika data yang ingin ditampilkan dimulai dari baris ke-n sampai ke akhir file.

5) line

Bentuk umum:

line [< file]

Digunakan untuk membaca baris pertama dari standar input ke standar output.

Contoh penulisan : **line < file** sama dengan **head -1 file**.

6) cmp

Bentuk umum:

cmp [-l | -s] file1 file2

Digunakan untuk membandingkan isi file1 dengan file2, terutama file biner. Tidak akan memberikan keluaran apapun jika kedua file yang dibandingkan ternyata sama. Jika file1 dan file2 berbeda, maka karakter/byte dan nomor baris yang berbeda ditampilkan. Option **-s** digunakan untuk memperoleh nilai exit saja (0 jika sama dan 1 jika tidak sama) dan option **-l** digunakan untuk menampilkan nomor baris dalam bentuk desimal dan karakter kedua file yang berbeda dalam kode oktal.

7) diff

Bentuk umum:

diff [-b | -e | -f | -h] file1 file2

Digunakan untuk memberitahukan perbedaan dari dua file teks, dimana pembandingan dilakukan perbaris.

Setelah perintah ini digunakan, ada beberapa karakter yang perlu diperhatikan. Di antaranya adalah < (menyatakan file1), > (menyatakan file2), a (append), c (change) dan d (delete). Option:

- **b**, Untuk mengabaikan seluruh karakter whitespace (spasi dan tab)
- **e**, Digunakan untuk membangkitkan perintah a, c, atau d guna menciptakan file2 dari file1
- **f**, Untuk menghasilkan perintah dalam urutan terbalik.
- **h**, Untuk file yang berukuran besar.

8) diff3

Bentuk umum:

diff3 [-e | -x] file1 file2 file3

Digunakan untuk membandingkan isi dari 3 file dan memberitahukan perbedaannya ke standar output.

Setelah perintah ini dijalankan ada beberapa karakter yang perlu diperhatikan, yaitu: ==== (ketiga file berbeda), ====n (file ke-n berbeda)

Setelah itu ada format yang perlu diperhatikan, yaitu:

a) f:n a, teks perlu ditambahkan setelah baris n pada file f.

b) f:n1 n2 c, teks perlu diubah dari baris n1 sampai n2 pada file f.

Option:

- **e**, keluarannya berisi langkah untuk mengubah file1 dan file2 agar menjadi file3 (hanya untuk pembandingan file yang menghasilkan kode "====" atau "====3".
- **x**, serupa dengan -e namun hanya berlaku khusus untuk kondisi "===="

9) comm3

Bentuk umum:

comm [-123] file1 file2

Digunakan untuk memberitahukan perbedaan atau persamaan baris dari dua buah file.

File1 dan file2 merupakan dua buah file yang telah diurutkan. Jika tanpa option, laporan yang dihasilkan dalam bentuk tiga kolom (kolom ke-1 memberitahukan isi yang hanya terdapat pada file1 dan kolom ke-2 pada file2 dan kolom ke-3 adalah isi yang terdapat pada file1 dan file2). Apabila option disebutkan, maka kolom yang bersesuaian dengan option tidak akan ditampilkan.

10) grep (*global regular expression printer)*

Bentuk umum:

grep [option] 'regular_ekspresi' [file…]

Digunakan untuk mencari string pada suatu file dengan melibatkan regular ekspresi.

Regular ekspresi yang ada:

a) . Sebarang karakter tunggal.

b) ***** Nol atau sejumlah karakter apa saja.

c) **[]** Sebuah karakter yang terletak di dalam tanda [].

d) **[^…]** Selain karakter yang terletak sesudah tanda ^ dan sebelum].

e) **^** Terletak di awal baris.

f) **$** Terletak di akhir baris.

g) **(expr\\)** Cocok dengan ekspresi expr.

h) **n** Cocok dengan n ekspresi (n berupa bilangan bulat).

11) fgrep *(fixed grep atau fast grep)*

Bentuk umum:

```
fgrep [ option ] [ file… ]
```

Digunakan untuk mencari beberapa string tetapi pemakaian regular ekspresi tidak didukung.

Di antara keluarga grep, fgrep merupakan program yang menggunakan memori paling sedikit.

Pencarian beberapa string harus diletakkan pada baris yang berlainan. Dimana target awal diawali dengan ‘ atau “ dan target terakhir diakhiri dengan ‘ atau “.

12) egrep *(extended grep)*

Bentuk umum:

```
egrep [ option ]  ‘regular_ekspresi’ [ file… ]
```

Digunakan untuk mencari beberapa string dan dapat melibatkan regular ekspresi.

Perintah ini menggunakan memori lebih banyak daripada dua perintah lainnya, namun memberikan kecepatan yang paling tinggi.

Regular ekspresi \(dan \) tidak ada pada egrep.

Regular ekspresi tambahan:

a) **+** Cocok dengan satu atau beberapa karakter yang terletak di depan tanda ini.

b) **?** Cocok dengan nol atau beberapa karakter yang terletak di depan tanda ini.

c) **|** atau.

d) **( )** Penggolongan.

13) sort

Bentuk umum:

sort [option] [file…]

Digunakan untuk mengurutkan data yang terdapat pada satu atau beberapa file, dan hasilnya diletakkan ke standar output. File yang akan diurutkan memiliki organisasi seperti: record, field, dan kunci. Secara default, hasil pengurutan dalam bentuk ascending.

14) cut

Bentuk umum:

cut -c*list* [file…]
cut -*list* [-d char] [-s] [file…]

Digunakan untuk mengambil data tertentu berdasarkan kolom atau field dari suatu file.

Beberapa perintah pada cut :

- list digunakan untuk menyatakan range, seperti:1,4,7,1-3.
- **c*list*** digunakan jika kolom yang dijadikan sebagai patokan pengambilan data.
- ***list*** digunakan jika data yang diambil ditampilkan perbaris.
- Option:
  - d diikuti sebuah karakter menyatakan karakter yang dipakai sebagai pembatas antarfield, jika tidak disertakan maka karakter tab yang dijadikan sebagai pemisah field.
  - s akan mengabaikan baris yang tidak punya pembatas.

15) paste

Bentuk umum:

paste [-s] [-d*list*] [file…]

Digunakan untuk merekatkan sejumlah baris dari beberapa file, yang dilakukan baris perbaris. Baris pertama file ke-1 direkatkan dengan baris pertama file ke-2.

Option:

- s, digunakan untuk menggabungkan isi file secara sekuensial, **-d*list***, lihat perintah cut.

P2.2. Contoh Kasus

Buat program dengan editor vi untuk menghasilkan tampilan berikut ini :

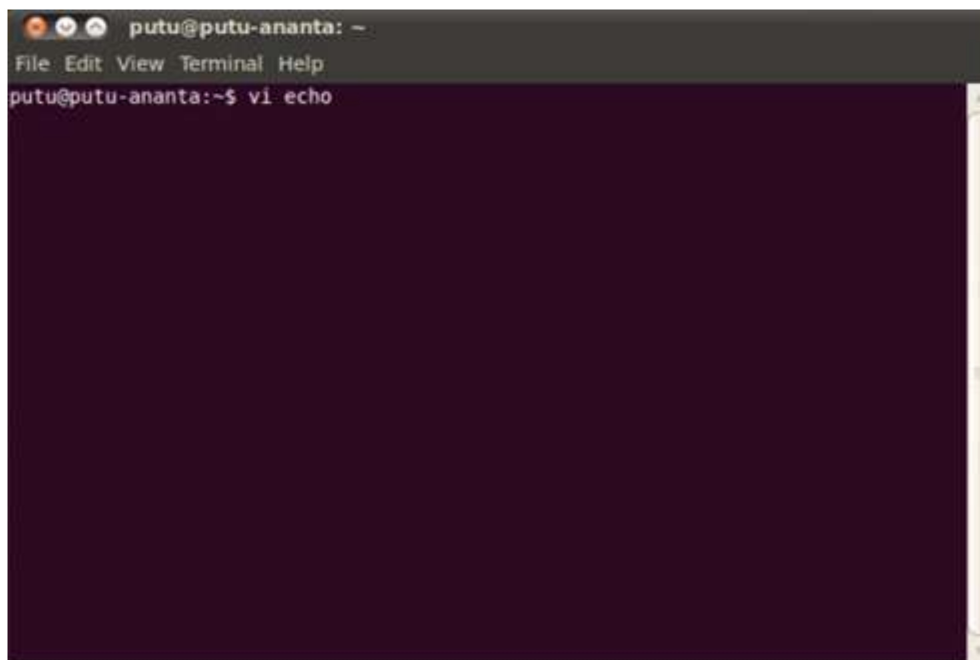
Nama Anda : $nama

Hobi Anda : $hobi

Saya $nama, Hobi saya $hobi sampai jumpa

Langkah pengerjaan :

1. Log In ke linux anda
2. Setelah masuk pada desktop linux, buka terminal dengan cara CTRL+ALT+T atau dengan melakukan pilihan Application – Accessories – Terminal
3. Lalu buka vi editor dengan ketik vi<spasi> nama file



4. Pada saat masuk editor vi maka anda berada pada mode perintah dan untuk mengetikkan perintah dibawah ini maka anda harus masuk ke mode penyisipan dengan tekan hutuf “I” pada keyboard untuk masuk ke mode penyisipan.
5. Ketikan program dibawah ini dan simpan program, untuk menjalankannya keluar dari editor vi menuju ke terminal dengan cara tekan ESC serta ketik : wq.

```
putu@putu-ananta: ~
File  Edit  View  Terminal  Help
echo "Nama anda :"
read nama
echo "Hobi Anda :"
read hobi
echo "Saya $nama,Hobi saya$hobi sampai jumpa."

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```

6. Kemudian di terminal anda ketikkan chmod<spasi>755<spasi>nama file untuk memberi hak akses dan ketik ./<nama file> untuk menjalankan program anda.

```
putu@putu-ananta: ~
File  Edit  View  Terminal  Help
putu@putu-ananta:~$ vi echo
putu@putu-ananta:~$ chmod 755 echo
putu@putu-ananta:~$ ./echo
```

Masukan nama dan hobi sesuai dengan yang anda inginkan maka akan tampil seperti contoh diatas.

P2.1. Latihan

Membuat program seperti dibawah ini menggunakan editor vi !

```
putu@putu-ananta: ~
File  Edit  View  Terminal  Help
x=100
y=25

let jumlah=$x+$y
let kurang=$x-$y
echo "hasil penjumlahan=$jumlah"
echo "hasil pengurangan=$kurang"

"operasi" 9 lines, 114 characters
```

Langkah pengerjaan :

1. Buka terminal dengan klik Applications—Accesories—Terminal atau tekan CTRL+ALT+T
2. Lalu masuk ke vi editor dengan ketik vi<spasi>operasi lalu tekan huruf “i” pada keyboard dan ketikkan :

   X = 100
   Y = 25
   let jumlah = $X + $Y
   let kurang = $X - $Y
   echo “hasil penjumlahan = $jumlah”
   echo “hasil pengurangan = $kurang”
3. Lalu simpan dan keluar kembali ke terminal dengan tekan ESC dan ketik :wq
4. Lalu di terminal beri hak akses dengan ketik chmod<spasi>755<spasi> operasi lalu ketik **./operasi** untuk menjalankan program

```
putu@putu-ananta: ~
File  Edit  View  Terminal  Help
putu@putu-ananta:~$ vi operasi
putu@putu-ananta:~$ chmod 755 operasi
putu@putu-ananta:~$ ./operasi
```

Masukan sebuah nilai untuk X dan Y maka perhatikan hasil dari program tersebut.

P2.2. Daftar Pustaka

Kadir, Abdul.*Pengenalan Unix dan Linux.*Penerbit Andi Yogyakarta.2002.

Sidik, Betha. *Panduan Bekerja dalam Lingkungan Unix dan Linux*, Penerbit Informatika. 2004.